

INITIAL REPORT

TEAM BRISK

1. Project Description

1.1. Aims for the Project. In recent years, so many traffic simulation systems have been developed to analyze the traffic problem. However, those systems are not suitable or practical in some situations. Our team aims at developing a traffic simulation system that can support different types of vehicles running on several kinds of roads under some traffic management policies. The system will be of high quality and completed in time.

1.2. Strategy for Achieving Aims. According to the aims, our group chose the strategy, development platform and language to develop the project. Overall, we follow the waterfall model to develop our project step by step.

Strategy: The strategy of group project is waterfall model that we can develop the group project step by step. In order to finish our project in time, we shall plan well and follow the timetable strictly. We will also use development techniques that we are familiar with and set priorities to different functional requirements and implement the compulsory ones first. To make the quality better, we plan to spend more time designing before coding as well as apply testing techniques such as unit testing. We also plan to hold design review and code review meetings.

1.3. Initial Plan. Our group designed the initial timetable as a schedule of our group project. Table 1 is the initial plan of time.

Phases	Start date	End date
Planning	2015-01-26	2015-01-28
Requirement analysis	2015-01-29	2015-02-02
Architecture design	2015-02-03	2015-02-07
UI design	2015-02-03	2015-02-08
Detail design	2015-02-09	2015-02-19
Implementation	2015-02-20	2015-03-15
Testing	2015-03-16	2015-03-20

TABLE 1. Initial Time Table

1.4. Requirements Analysis. Requirements Analysis is one of the most essential parts of the model. It is our obligation to know what the requirements are. According to the requirements, we can design the model correctly.

System context: The system context of this system is to simulate a road network which contains several different parts including: roundabouts and multi-lane junctions. There are also several types of vehicles running on it under some traffic policies.

Functional requirements:

1. Traffic model should have different vehicles (cars, coaches, buses etc.). (High Priority)

2. Traffic model can be operated in different parts of road (eg. straight roads, curves, roundabouts). (High Priority)

3. Designing suitable maps. (High Priority)

4. Model should be based on time to change. (High Priority)

5. Considering emergency situation. (High Priority)

6. Allowing different traffic management policy. (Medium Priority)

7. Comparing different policies. (Low Priority)

Non-functional requirements:

1. Efficiency: space and time complexity should not be very high. (Mandatory)

2. Flexible: should fit with general situations not only the situation we designed. (Optional)

3. Standards requirements: should fit with software engineering principle. (Optional)

4. Implementation requirements: based on Java programming language. (Optional)

5. Reliability: can run continuously for at least 5 hours. (Mandatory)

6. Robustness: can deal with unexpected input. (Mandatory)

7. Usability: can be used without training. (Optional)

1.5. Designing. The Business Concept Model shows as figure 1.

The Use Case shows as figure 2.

The Initial System Architecture shows as figure 3.

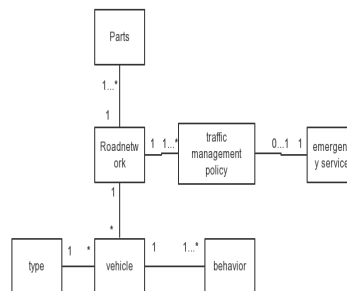


FIGURE 1. Business Concept Model

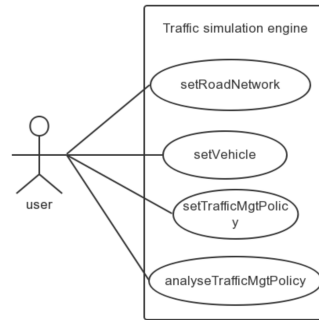


FIGURE 2. Use Case

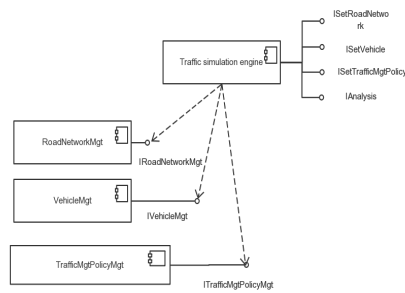


FIGURE 3. Initial System Architecture

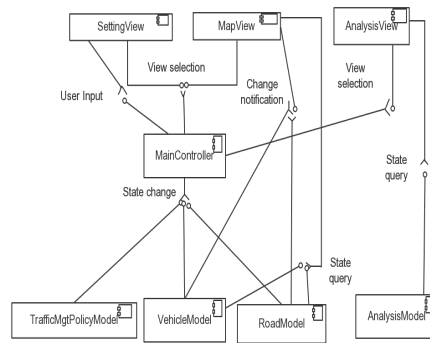


FIGURE 4. MVC Style Architecture

The MVC Style Architecture shows as figure 4.

1.6. Construction. Our group synchronized the code by Github and use the MVC design as pattern to improve code quality. Considering the unity, our group is trying to uniform some code styles by designing coding standards.

1.7. **Testing and debugging.** A special member responsible for testing and debugging was assigned by the group.

1.8. **Development Platform:Eclipse.**

1.9. **Development Language:JAVA.**

2. Project Organisation

2.1. **Team Members and Roles.** Our group designed the team members and roles table as the workload of each member. Table 2 is the Team Members and Roles.

Member	Roles and Responsibilities
Wang,Shengsheng	Coordinator,Architect and Developer
Qu,Tong	Model builder and Developer
Liu,Yiran	GUI designer and Developer
Yamani,Lujain Bassam	Quality assurance and Developer
Duan,Zhaoxu	Developer and Document writer

TABLE 2. Team Members and Roles

2.2. **Peer Assessment.** 1. 20(100/5) marks are allocated to each member.

2. The basic marks for everyone is 90% of the total, that is $20 * 0.9 = 18$.

3. $2 * 5 = 10$ marks are put in the mark pool and will be distributed in the end of the project according to contribution and overall performance.(by secret voting)

4. There are some cases in which one might lose marks and the marks will be allocated to mark pool.

a.Do not present in the meeting(-2).

b.Being late for the meeting for more than 5 minutes(-1)

c.Fail to finish tasks in time(-2 to -5)

2.3. **Conflicts Resolve.** There will be a group meeting for every week.We will share work by group members through Github. In case of a conflict, the group member in charge of the specific part will be required to design an approach to resolve the problem. However, there will be a group communication before addressing the conflict.